

**WHAT IS CLAIMED IS:**

1. A method comprising:  
 providing a computer including  
 a processor and  
 a memory operably coupled to the processor;  
 providing a first software program capable of being operably installed on the  
 computer;  
 providing a second software program  
 capable of being operably installed on the computer and  
 capable of being used interoperably with the first software program;  
 modifying the second software program to include data defining a specific point in  
 time after which the second software program cannot be used interoperably with the first  
 software program;  
 digitally signing the second software program including the data defining the specific  
 point in time;  
 determining whether the second software program has been altered after the digitally  
 signing;  
 verifying that the specific point in time has not passed; and  
 using the second software program interoperably with the first software program if  
 and only if  
 the determining determines that the second software program has not been  
 altered after the digitally signing and  
 the verifying verifies that the specific point in time has not passed.

1           2.       The method of claim 1, wherein  
2           the second software program includes a device information file and  
3           the data defining the specific point in time is included in the device information file.

1           3.       The method of claim 1, further comprising  
2           verifying after the using that the specific point in time has not passed and  
3           blocking interoperable use of the second software program with the first software  
4           program if the specific point in time has passed.

1           4.       The method of claim 1, wherein  
2           the first software program is an operating system and  
3           the second software program is an application software program.

1           5.       The method of claim 1, wherein  
2           the first software program is an operating system and  
3           the second software program is a peripheral driver.

1           6.       The method of claim 1, wherein  
2           the first software program is an application software program and  
3           the second software program is a plug-in.

1 7. A computer system comprising:  
2 a processor;  
3 a first software program capable of being operably coupled to the processor;  
4 a digitally signed second software program, the second software program  
5 capable of being operably coupled to the processor,  
6 capable of being used interoperably with the first software program, and  
7 including data defining a specific point in time after which the second  
8 software program cannot be used interoperably with the first software  
9 program; and  
10 a memory coupled to the processor, the memory including  
11 means for determining whether the second software program has been altered,  
12 means for verifying that the specific point in time has not passed, and  
13 means for using the second software program interoperably with the first  
14 software program if and only if  
15 it is determined that the second software program has not been altered  
16 and  
17 it is verified that the specific point in time has not passed.

1 8. The computer system of claim 7, wherein  
2 the second software program includes a device information file and  
3 the data defining the specific point in time is included in the device information file.

1 9. The computer system of claim 7, wherein the memory coupled to the  
2 processor further includes  
3 means for verifying after an interoperable use of the second software program with  
4 the first software program that the specific point in time has not passed and  
5 means for blocking interoperable use of the second software program with the first  
6 software program if the specific point in time has passed.

1 10. The computer system of claim 7, wherein  
2 the first software program is an operating system and  
3 the second software program is an application software program.

1 11. The computer system of claim 7, wherein  
2 the first software program is an operating system and  
3 the second software program is a peripheral driver.

1 12. The computer system of claim 7, wherein  
2 the first software program is an application software program and  
3 the second software program is a plug-in.

1 13. An apparatus for limiting use of a first software program interoperably with a  
2 second software program comprising:

3 means for modifying the second software program to include data defining a specific  
4 point in time after which the second software program cannot be used  
5 interoperably with the first software program;

6 means for digitally signing the second software program including the data defining  
7 the specific point in time;

8 means for determining whether the second software program has been altered after the  
9 digitally signing;

10 means for verifying that the specific point in time has not passed; and

11 means for using the second software program interoperably with the first software  
12 program if and only if

13 it is determined that the second software program has not been altered after the  
14 digitally signing and

15 it is verified that the specific point in time has not passed.

1 14. The apparatus of claim 13, further comprising:

2 means for verifying after an interoperable use of the second software program with  
3 the first software program that the specific point in time has not passed and

4 means for blocking interoperable use of the second software program with the first  
5 software program if the specific point in time has passed.

1        15.     The apparatus of claim 13, wherein  
2        the second software program includes a device information file and  
3        the data defining the specific point in time is included in the device information file.

1        16.     The apparatus of claim 13, wherein  
2        the first software program is an operating system and  
3        the second software program is an application software program.

1        17.     The apparatus of claim 13, wherein  
2        the first software program is an operating system and  
3        the second software program is a peripheral driver.

1        18.     The apparatus of claim 13, wherein  
2        the first software program is an application software program and  
3        the second software program is a plug-in.

1        19.    A method comprising:  
2        providing a computer including  
3                a processor and  
4                a memory operably coupled to the processor;  
5        providing an application software program capable of being operably installed on the  
6 computer;  
7        providing a plug-in  
8                capable of being operably installed on the computer and  
9                capable of being used interoperably with the application software program;  
10        modifying the plug-in to include a specific set of preconditions limiting use of the  
11 plug-in interoperably with the application software program;  
12        digitally signing the plug-in including the specific set of preconditions;  
13        determining whether the plug-in has been altered after the digitally signing;  
14        verifying that the specific set of preconditions limiting use of the plug-in  
15 interoperably with the application software program is met; and  
16        using the plug-in interoperably with the application software program if and only if  
17                the determining determines that the plug-in has not been altered after the  
18                digitally signing and  
19                the verifying verifies that the specific set of preconditions is met.

1        20.    The method of claim 19, wherein the specific set of preconditions limiting use  
2 of the second software program interoperably with the first software program includes data  
3 defining a specific point in time after which the second software program cannot be used  
4 interoperably with the first software program.

1        21.     The method of claim 19, wherein  
2        the second software program includes a device information file and  
3        the data defining the specific point in time is included in the device information file.

1        22.     The method of claim 19, further comprising  
2        verifying after the using that the specific set of preconditions limiting use of the  
3        second software program interoperably with the first software program continues to be met  
4        and  
5        blocking interoperable use of the second software program with the first software  
6        program if any of the specific set of preconditions limiting use are not met.

1        23.     A computer system comprising:  
2        a processor;  
3        a first software program capable of being operably coupled to the processor;  
4        a digitally signed second software program, the second software program  
5                capable of being operably coupled to the processor,  
6                capable of being used interoperably with the first software program, and  
7                including data defining a specific point in time after which the second  
8                software program cannot be used interoperably with the first software  
9                program; and  
10       a memory coupled to the processor, the memory including  
11               a circuit for determining whether the second software program has been  
12               altered,  
13               a circuit for verifying that the specific point in time has not passed, and  
14               a circuit for using the second software program interoperably with the first  
15               software program if and only if  
16               the circuit for determining determines that the second software  
17               program has not been altered and  
18               the circuit for verifying verifies that the specific point in time has not  
19               passed.

1           24.     The computer system of claim 23, wherein the memory coupled to the  
2 processor further includes  
3           a circuit for verifying after an interoperable use of the second software program with  
4           the first software program that the specific point in time has not passed and  
5           a circuit for blocking interoperable use of the second software program with the first  
6           software program if the specific point in time has passed.

1           25.     An apparatus for limiting use of a first software program interoperably with a  
2 second software program comprising:  
3           a circuit for modifying the second software program to include data defining a  
4           specific point in time after which the second software program cannot be used  
5           interoperably with the first software program;  
6           a circuit for digitally signing the second software program including the data defining  
7           the specific point in time;  
8           a circuit for determining whether the second software program has been altered after  
9           the digitally signing;  
10          a circuit for verifying that the specific point in time has not passed; and  
11          a circuit for using the second software program interoperably with the first software  
12          program if and only if  
13          the circuit for determining determines that the second software program has  
14          not been altered after the digitally signing and  
15          the circuit for verifying verifies that the specific point in time has not passed.

16          26.     The apparatus of claim 25, further comprising:  
17          a circuit for verifying after an interoperable use of the second software program with  
18          the first software program that the specific point in time has not passed and  
19          a circuit for blocking interoperable use of the second software program with the first  
20          software program if the specific point in time has passed.